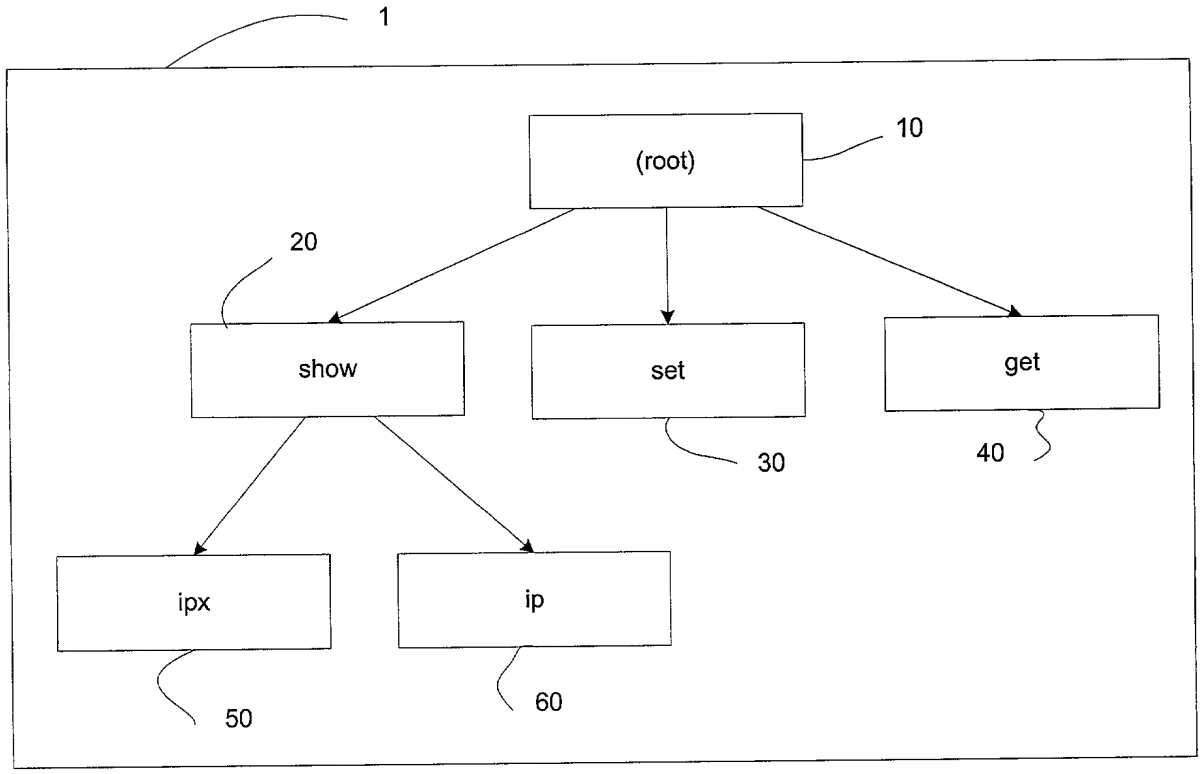

1
(root)
10
20
show
set
get
30
40
ipx
ip
50
60

Fig. 1

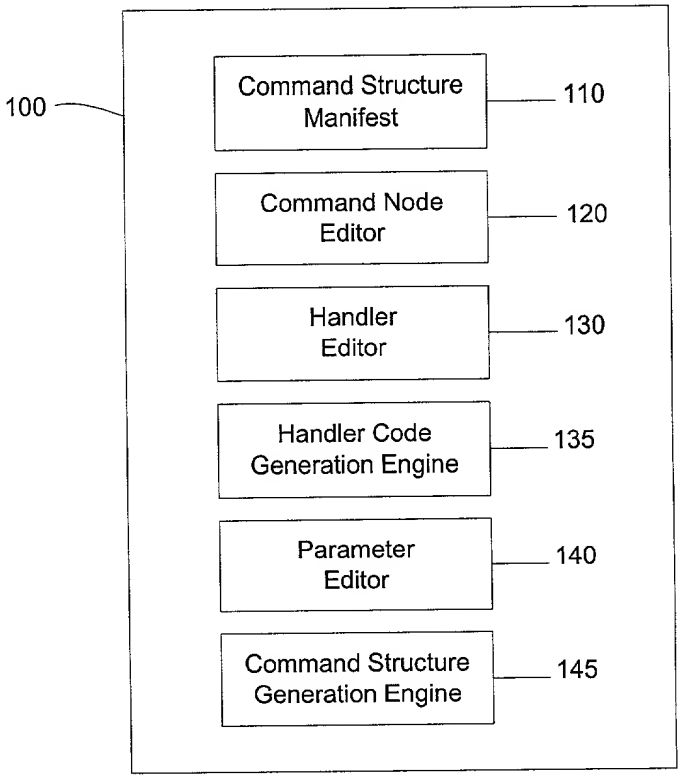

100
Command Structure Manifest
110
Command Node Editor
120
Handler Editor
130
Handler Code Generation Engine
135
Parameter Editor
140
Command Structure Generation Engine
145

Fig. 2

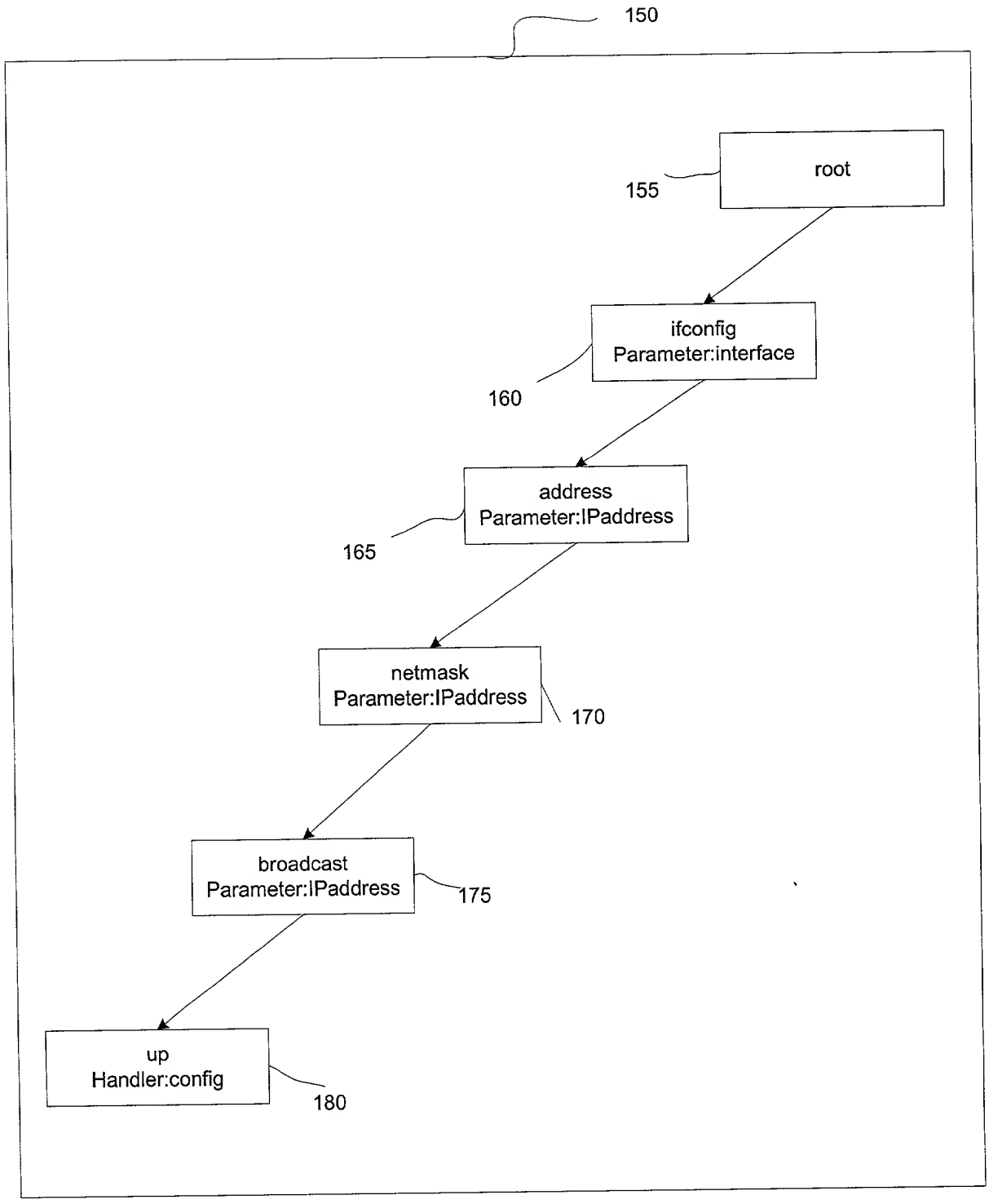

150
root
155
ifconfig
Parameter:interface
160
address
Parameter:IPaddress
165
netmask
Parameter:IPaddress
170
broadcast
Parameter:IPaddress
175
up
Handler:config
180

Fig. 3

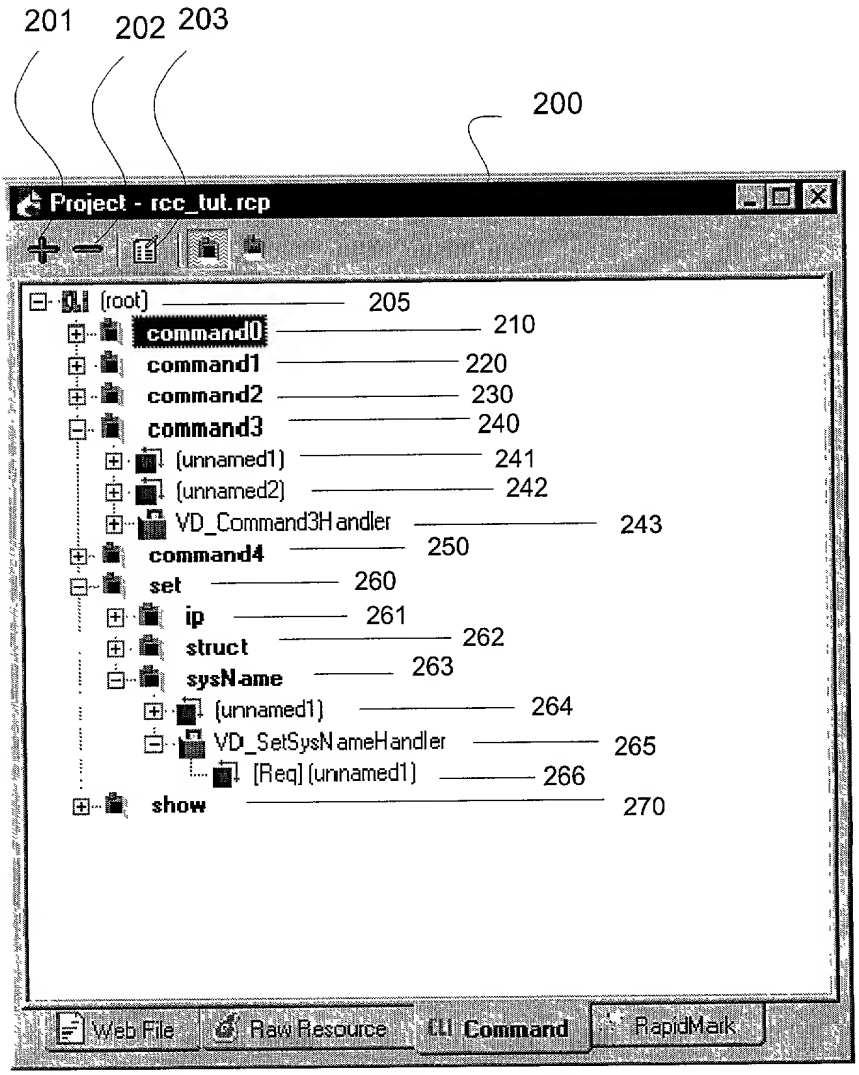

201
202
203
200
Project - rcc_tut.rcp
(root)
205
command0
210
command1
220
command2
230
command3
240
(unnamed1)
241
(unnamed2)
242
VD_Command3Handler
243
command4
250
set
260
ip
261
struct
262
sysName
263
(unnamed1)
264
VD_SetSysNameHandler
265
[Req] (unnamed1)
266
show
270
Web File
Raw Resource
CLI Command
RapidMark

Fig. 4

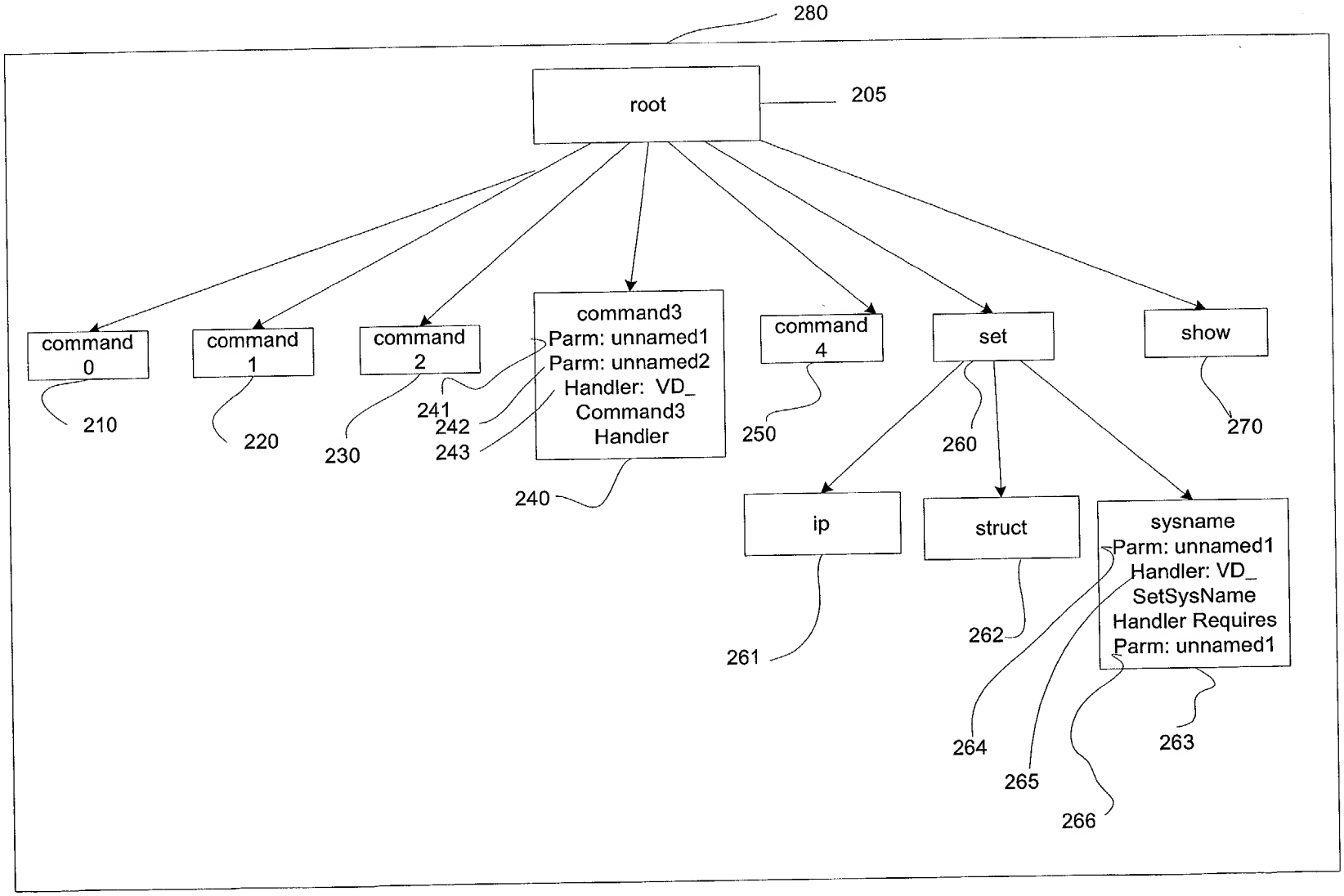

280
root
205
command 0
command 1
command 2
command3
Parm: unnamed1
Parm: unnamed2
Handler: VD_
Command3
Handler
command 4
set
show
210
220
230
241
242
243
240
250
260
270
ip
struct
sysname
Parm: unnamed1
Handler: VD_
SetSysName
Handler Requires
Parm: unnamed1
261
262
263
264
265
266

Fig. 5

Add a Command --300

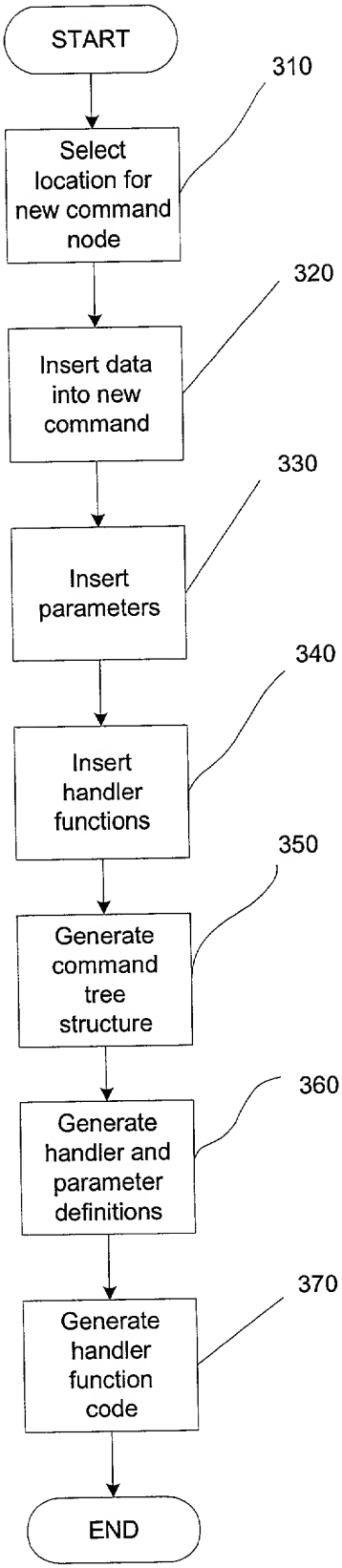


Fig. 6

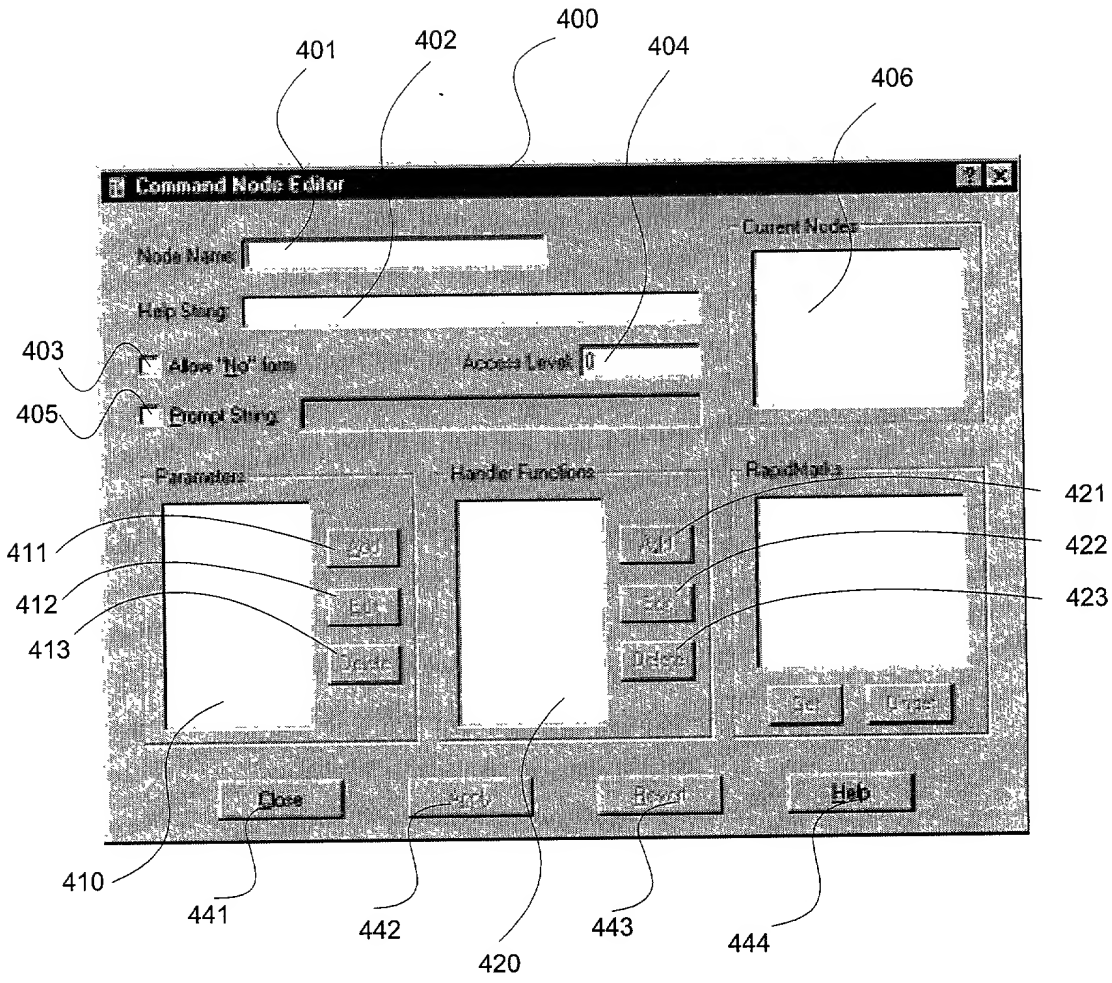

400
401
402
404
406
Command Node Editor
Node Name
Help String
Current Nodes
403
Allow "No" form
Access Level
0
405
Prompt String
Parameters
Handler Functions
RapidMotions
421
411
422
412
423
413
Close
Apply
Reset
Help
410
441
442
420
443
444

Fig. 7

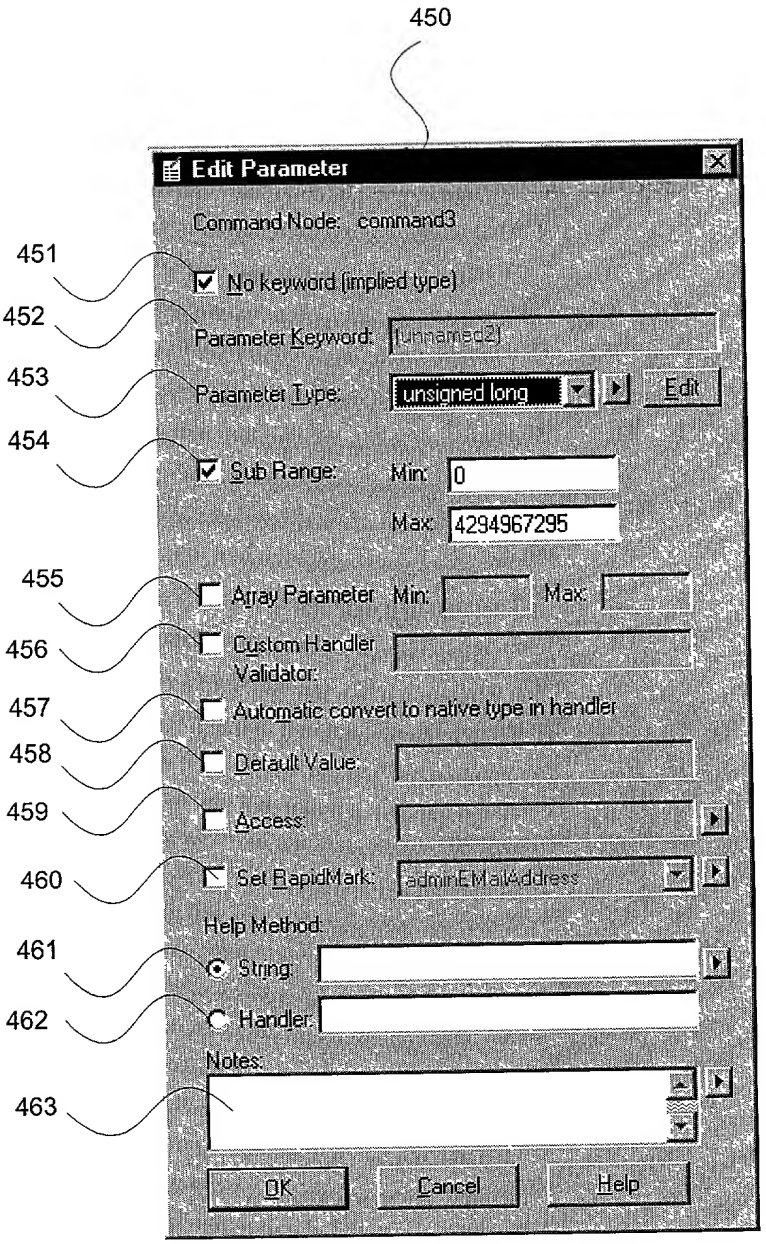

450
Edit Parameter
Command Node: command3
451
No keyword (implied type)
452
Parameter Keyword: [unnamed2]
453
Parameter Type: unsigned long
Edit
454
Sub Range: Min: 0
Max: 4294967295
455
Array Parameter Min:
Max:
456
Custom Handler Validator:
457
Automatic convert to native type in handler
458
Default Value:
459
Access:
460
Set RapidMark: adminEMailAddress
Help Method:
461
String:
462
Handler:
Notes:
463
OK
Cancel
Help

Fig. 8

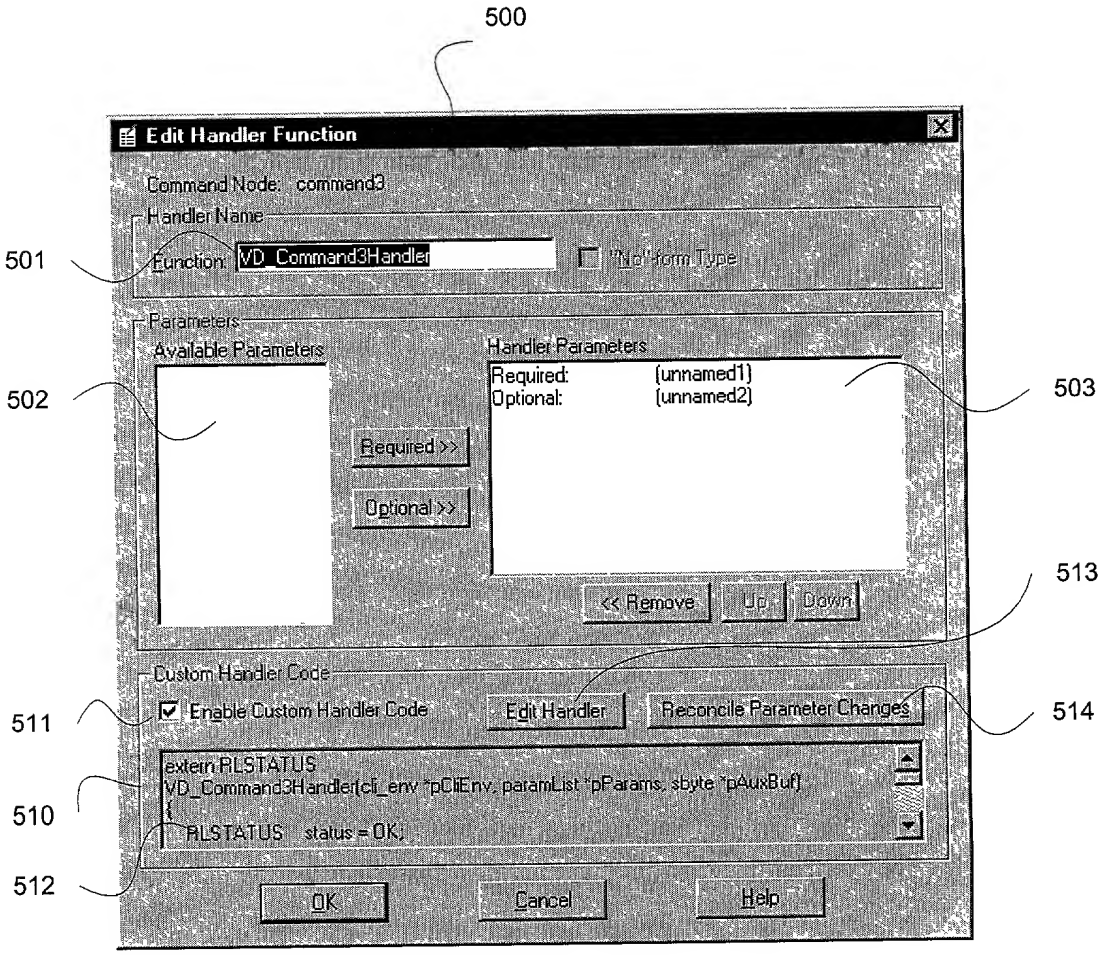

500
Edit Handler Function
Command Node: command3
Handler Name
Function: VD_Command3Handler
"No" form Type
Parameters
Available Parameters
Handler Parameters
Required: (unnamed1)
Optional: (unnamed2)
Required >>
Optional >>
<< Remove
Up
Down
Custom Handler Code
Enable Custom Handler Code
Edit Handler
Reconcile Parameter Changes
extern RLSTATUS
VD_Command3Handler(cli_env *pCliEnv, paramList *pParams, sbyte *pAuxBuf)
{
RLSTATUS status = OK;
OK
Cancel
Help
501
502
503
511
510
512
513
514

Fig. 9

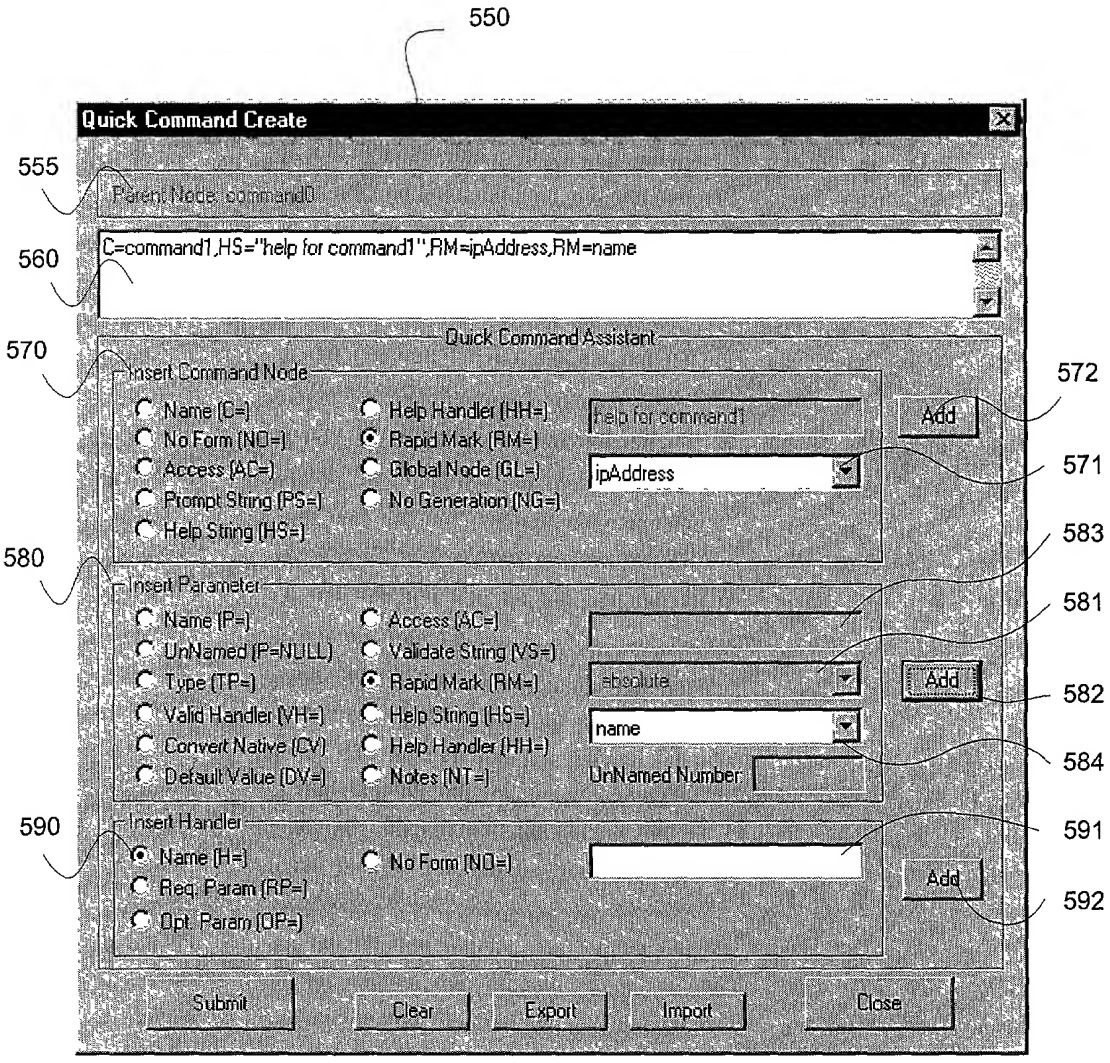

550
Quick Command Create
555
Parent Node: command0
560
C=command1,HS="help for command1",RM=ipAddress,RM=name
Quick Command Assistant
570
Insert Command Node
Name [C=]
No Form [NO=]
Access [AC=]
Prompt String [PS=]
Help String [HS=]
Help Handler [HH=]
Rapid Mark [RM=]
Global Node [GL=]
No Generation [NG=]
help for command1
ipAddress
572
Add
571
580
Insert Parameter
Name [P=]
UnNamed [P=NULL]
Type [TP=]
Valid Handler [VH=]
Convert Native [CV]
Default Value [DV=]
Access [AC=]
Validate String [VS=]
Rapid Mark [RM=]
Help String [HS=]
Help Handler [HH=]
Notes [NT=]
583
581
absolute
Add
582
name
584
UnNamed Number:
590
Insert Handler
Name [H=]
Req. Param [RP=]
Opt. Param [OP=]
No Form [NO=]
591
Add
592
Submit
Clear
Export
Import
Close

Fig. 10

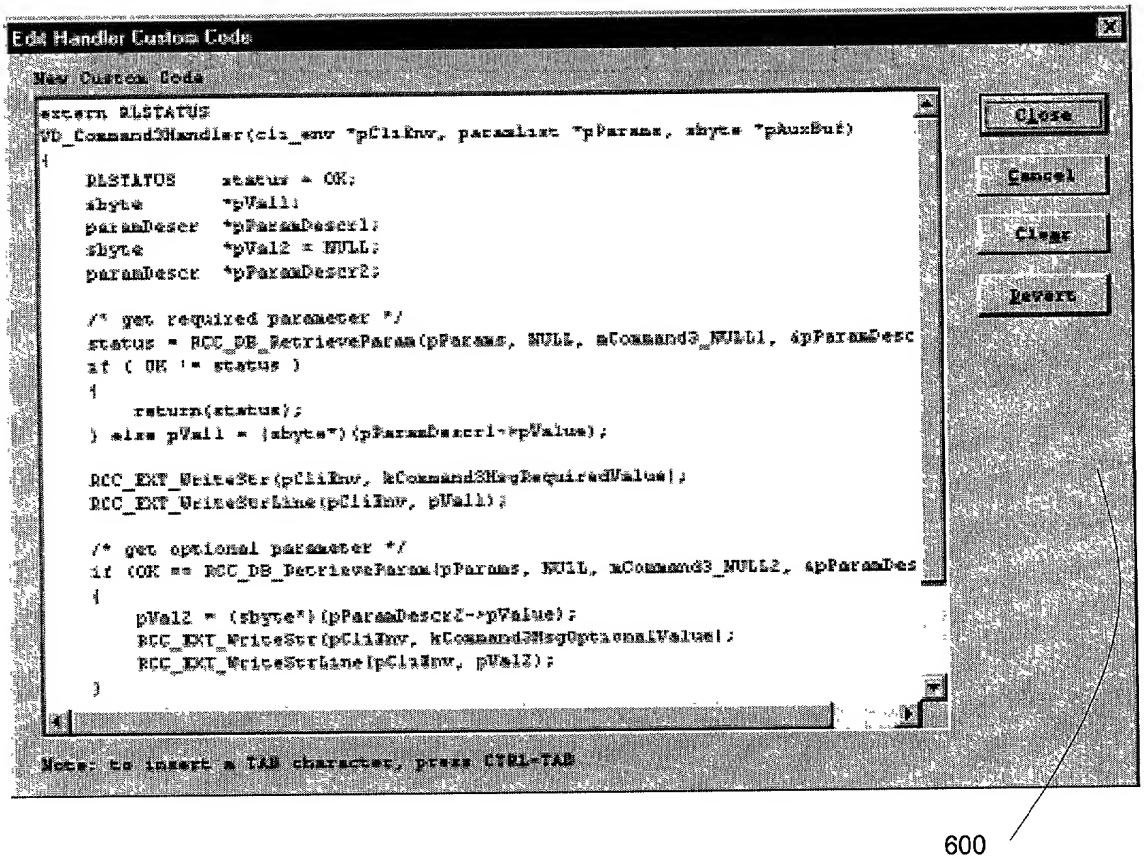

Edit Handler Custom Code
New Custom Code
extern RLSTATUS
VD_Command3Handler(cli_env *pCliEnv, paramlist *pParams, sbyte *pAuxBuf)
{
RLSTATUS status = OK;
sbyte *pVal1;
paramDescr *pParamDescr1;
sbyte *pVal2 = NULL;
paramDescr *pParamDescr2;
/* get required parameter */
status = RCC_DB_RetrieveParam(pParams, NULL, mCommand3_NULL1, &pParamDesc
if (OK != status)
{
return(status);
} else pVal1 = (sbyte*)(pParamDescr1->pValue);
RCC_EXT_WriteStr(pCliEnv, mCommand3MsgRequiredValue);
RCC_EXT_WriteStrLine(pCliEnv, pVal1);
/* get optional parameter */
if (OK == RCC_DB_RetrieveParam(pParams, NULL, mCommand3_NULL2, &pParamDes
{
pVal2 = (sbyte*)(pParamDescr2->pValue);
RCC_EXT_WriteStr(pCliEnv, mCommand3MsgOptionalValue);
RCC_EXT_WriteStrLine(pCliEnv, pVal2);
}
Note: to insert a TAB character, press CTRL-TAB
Close
Cancel
Clear
Revert
600

Fig. 11

Fig. 12

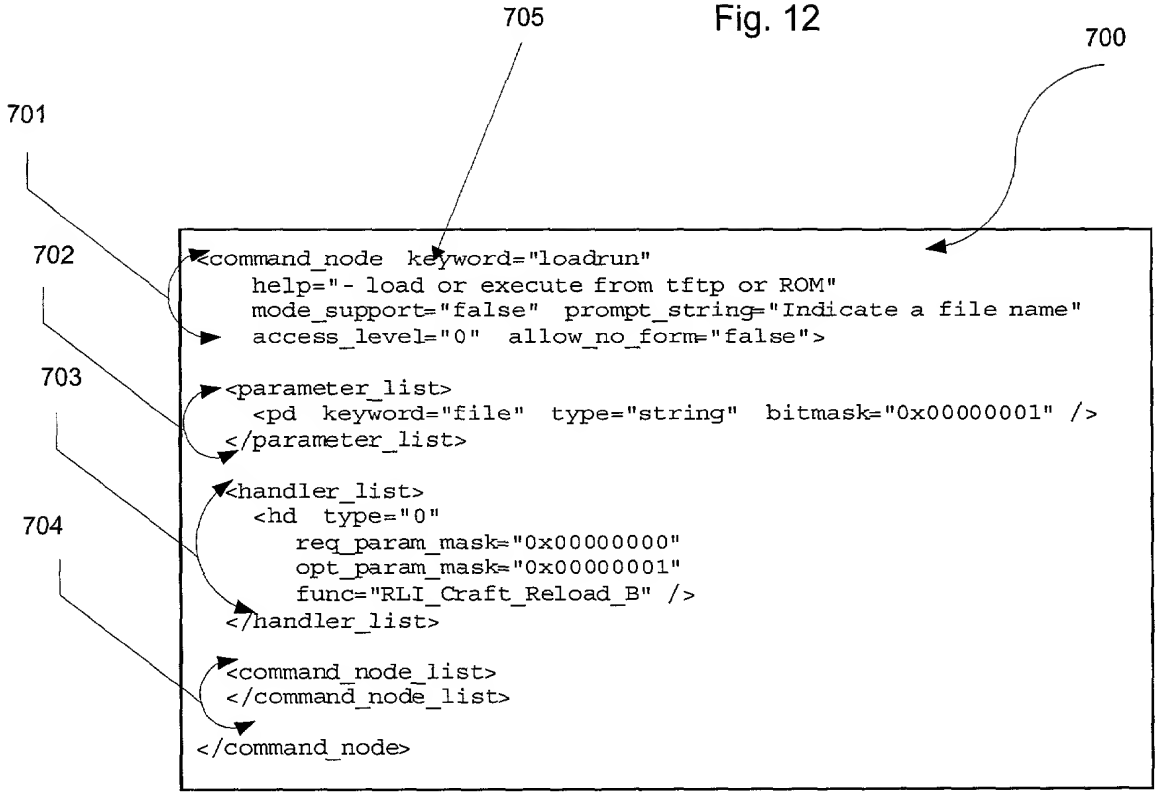

705
700
701
702
703
704
<command_node keyword="loadrun"
help="- load or execute from tftp or ROM"
mode_support="false" prompt_string="Indicate a file name"
access_level="0" allow_no_form="false">
<parameter_list>
<pd keyword="file" type="string" bitmask="0x00000001" />
</parameter_list>
<handler_list>
<hd type="0"
req_param_mask="0x00000000"
opt_param_mask="0x00000001"
func="RLI_Craft_Reload_B" />
</handler_list>
<command_node_list>
</command_node_list>
</command_node>

Fig. 13

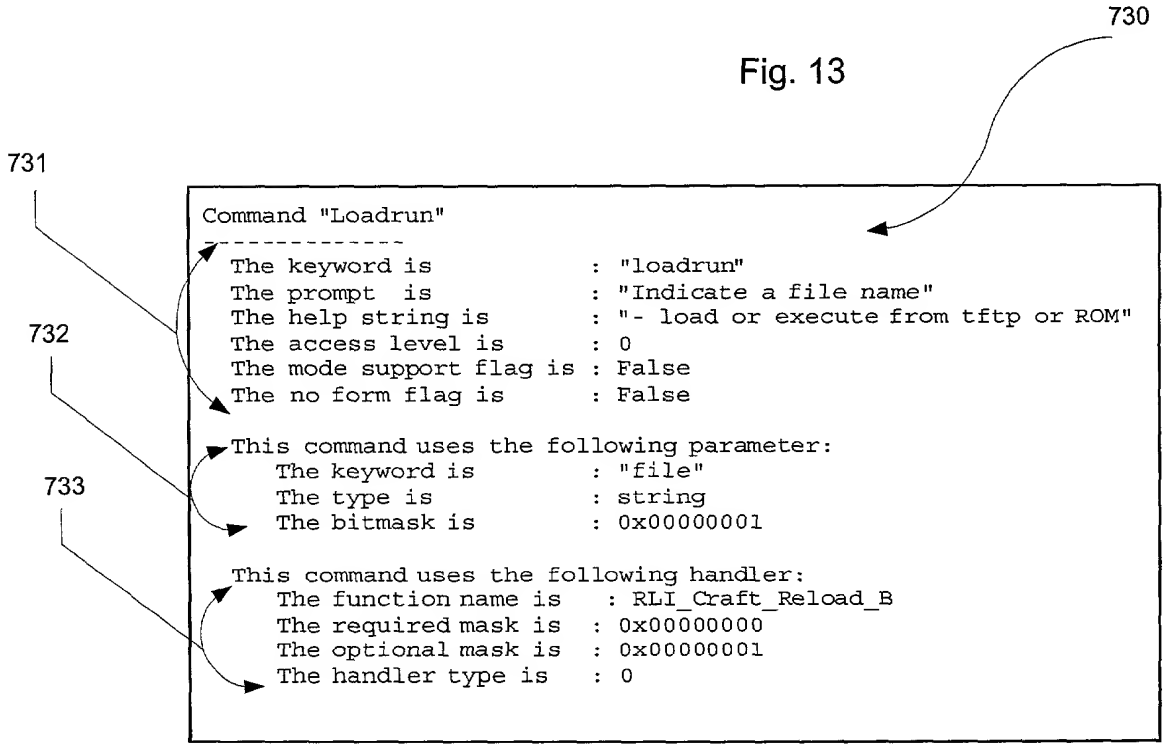
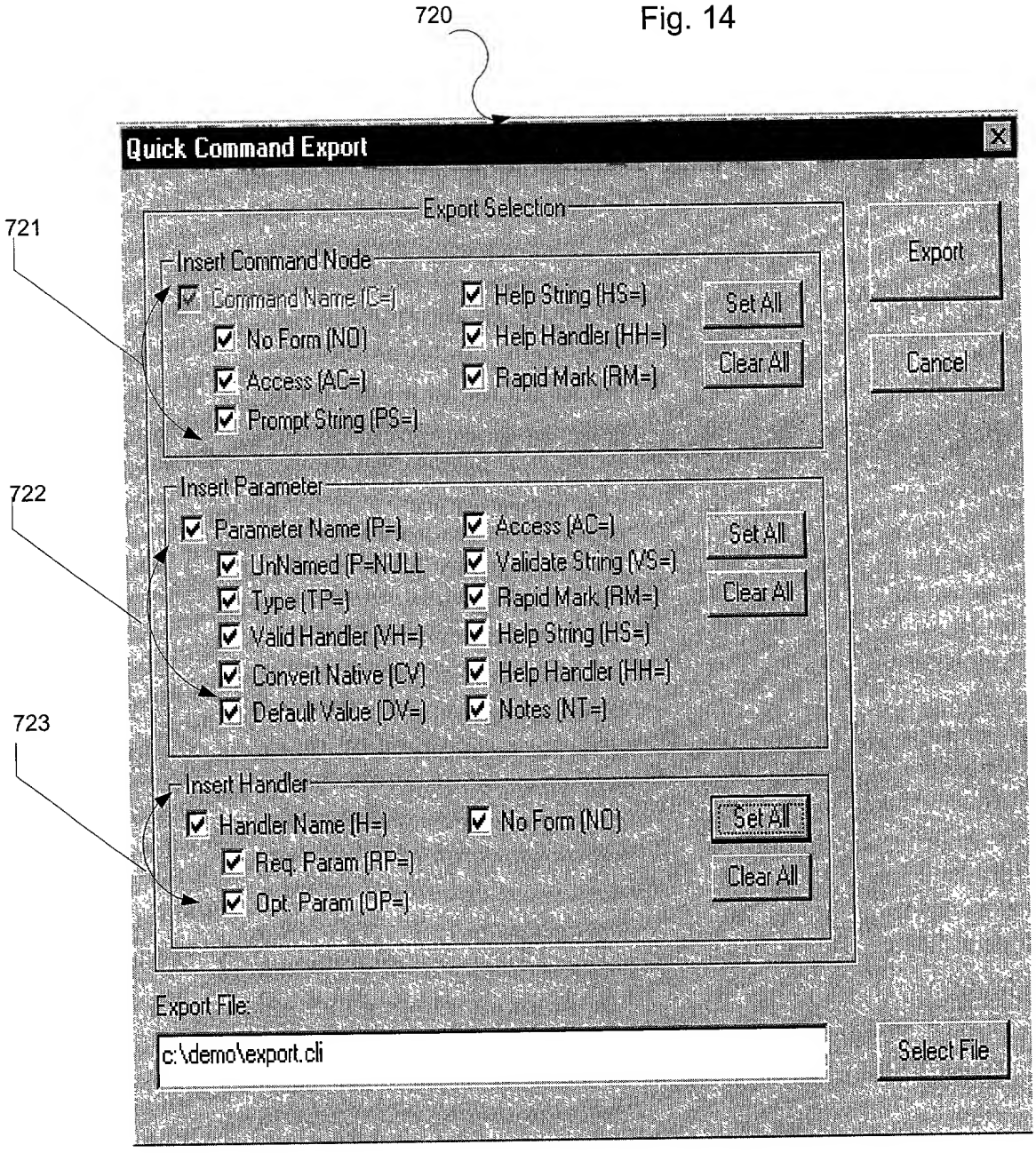

730
731
732
733
Command "Loadrun"
The keyword is : "loadrun"
The prompt is : "Indicate a file name"
The help string is : "- load or execute from tftp or ROM"
The access level is : 0
The mode support flag is : False
The no form flag is : False
This command uses the following parameter:
The keyword is : "file"
The type is : string
The bitmask is : 0x00000001
This command uses the following handler:
The function name is : RLI_Craft_Reload_B
The required mask is : 0x00000000
The optional mask is : 0x00000001
The handler type is : 0

Fig. 14


720
Quick Command Export
Export Selection
721
Insert Command Node
Command Name (C=)
No Form (NO)
Access (AC=)
Prompt String (PS=)
Help String (HS=)
Help Handler (HH=)
Rapid Mark (RM=)
Set All
Clear All
722
Insert Parameter
Parameter Name (P=)
UnNamed (P=NULL
Type (TP=)
Valid Handler (VH=)
Convert Native (CV)
Default Value (DV=)
Access (AC=)
Validate String (VS=)
Rapid Mark (RM=)
Help String (HS=)
Help Handler (HH=)
Notes (NT=)
Set All
Clear All
723
Insert Handler
Handler Name (H=)
Req. Param (RP=)
Opt. Param (OP=)
No Form (NO)
Set All
Clear All
Export
Cancel
Export File:
c:\demo\export.cli
Select File

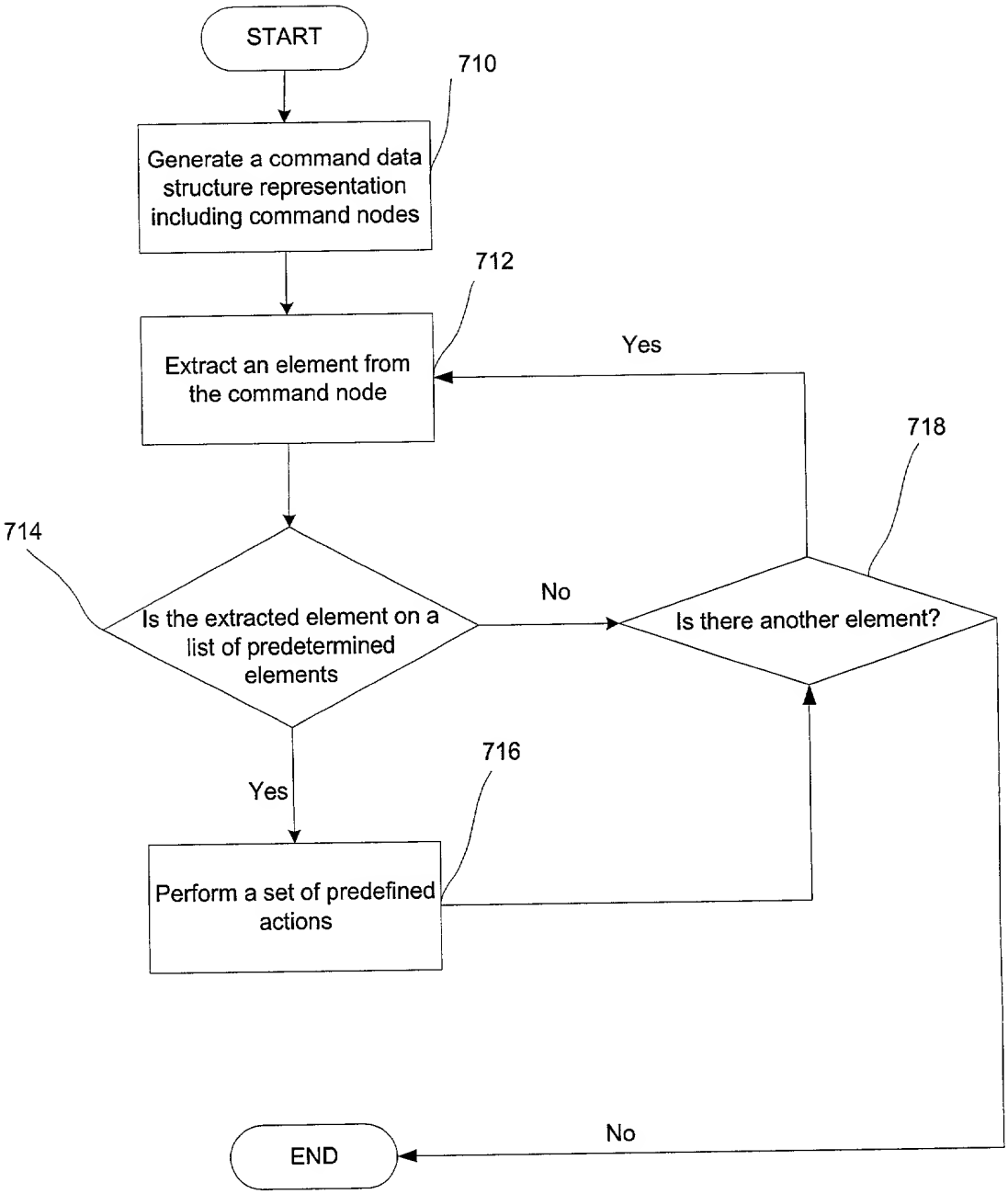

START
710
Generate a command data structure representation including command nodes
712
Extract an element from the command node
Yes
718
714
Is the extracted element on a list of predetermined elements
No
Is there another element?
Yes
716
Perform a set of predefined actions
No
END

Fig. 15